

**Theoretische Informatik II**  
**Übungen**

**Aufgabe 29.** Der Lambdaterm

$$\mathbf{pred} = \lambda n. \lambda f. \lambda x. n(\lambda u. \lambda v. v(uf))(\lambda w. x)(\lambda w. w)$$

implementiert die Vorgänger-Funktion auf den Churchnumerals.

- Was ist  $\mathbf{pred\ 0}$  ?
- Zeigen sie, dass für ein beliebiges positives Churchnumeral  $\mathbf{n} = \lambda f. \lambda x. f^n x$  gilt:

$$\mathbf{pred\ n} = \lambda f. \lambda x. f^{n-1} x$$

- Definieren sie mittels  $\mathbf{pred}$  einen Lambdaterm  $\mathbf{sub}$ , so dass für  $n > m$  gilt:

$$\mathbf{sub\ n\ m} = \lambda f. \lambda x. f^{n-m} x$$

**Aufgabe 30.** Zeige, dass  $SKK = I$ . Verwende dazu

- die Darstellung von  $S$ ,  $K$ , und  $I$  als Lambdaterme;
- die Funktionsgleichungen  $Sfgx = fx(gx)$ ,  $Kxy = x$ , und  $Ix = x$ .

**Aufgabe 31.** Zeige, dass mit  $T = \lambda xy. y(xxy)$  und  $\Phi = TT$  gilt: für alle  $f$  is  $f(\Phi f) = \Phi f$ . ( $\Phi$  is also ein Fixpunktkombinator). Warum ist es nicht möglich für die Variablen  $x$  und  $y$  geeignete Typen zu finden mit denen  $\Phi$  ein getypter Lambdaterm wird?

**Aufgabe 32.** Drücken Sie den Lambdaterm  $\lambda xy. yx$  nur mit den Kombinatoren  $S$  und  $K$  aus.

**Aufgabe 33.** Was versteht man unter

- Funktionaler Programmierung?
- Closure?
- $\alpha$ -,  $\beta$ - und  $\eta$ -Gleichheit?
- Freien Variablen, Variablenbindungen, und Variablenumgebungen?
- einem getypten Lambda-Kalkül. Was unterscheidet ihn von einem ungetypten Lambda-Kalkül? Ist Scheme getypt?
- Normalisierung?
- Normalform?

Geben Sie jeweils Beispiele an.

**Aufgabe 34.** Geben Sie einen Lambdaterm an, der keine Normalform besitzt. Können Sie einen getypten Lambdaterm angeben, der keine Normalform besitzt? Berechnen sie mit Churchnumeralen die Normalform von  $1 \cdot 1$ .